

Software Engineering Project

Name: Gagan Deep Singh (17BCI0140)

Team mate: Onkar Sharma (17BCE0276)

Project Abstract: Weather Data Analyzer (Weather Forecasting)

08|01|2019

This project is particular to data analysis, analyzing weather data like temperature, humidity and wind to predict the state of the atmosphere for a given location. This is basically achieved by observing patterns in events (patterns recognition). A graphical user interface is provided to users (including administrator), along with user ID and password. User (casual or end users) will enter current temperature, humidity and wind (Direction and speed), system will take these parameters and with the help of previous data in database, will predict weather for next day or later in that day. Current data entered by user if available will be stored in database (Data collection), to improve future predictions. Only system administrator has the authorization to add or remove data strictly based on standard criteria. System administrator can also modify data only in case of data entered in correctly and is essential to the system, otherwise incorrect data can only be deleted. The system has basically only these two types of users.

Applications of this project are in fields of agriculture, forestry and general predictions for citizens. As this project' predictions are based only on limited parameters and omitted infeasible computation for regular computer systems, the software can be proved unreliable for military, navy, Air Traffic Control and marine applications. But will have the provision to advance to such applications.

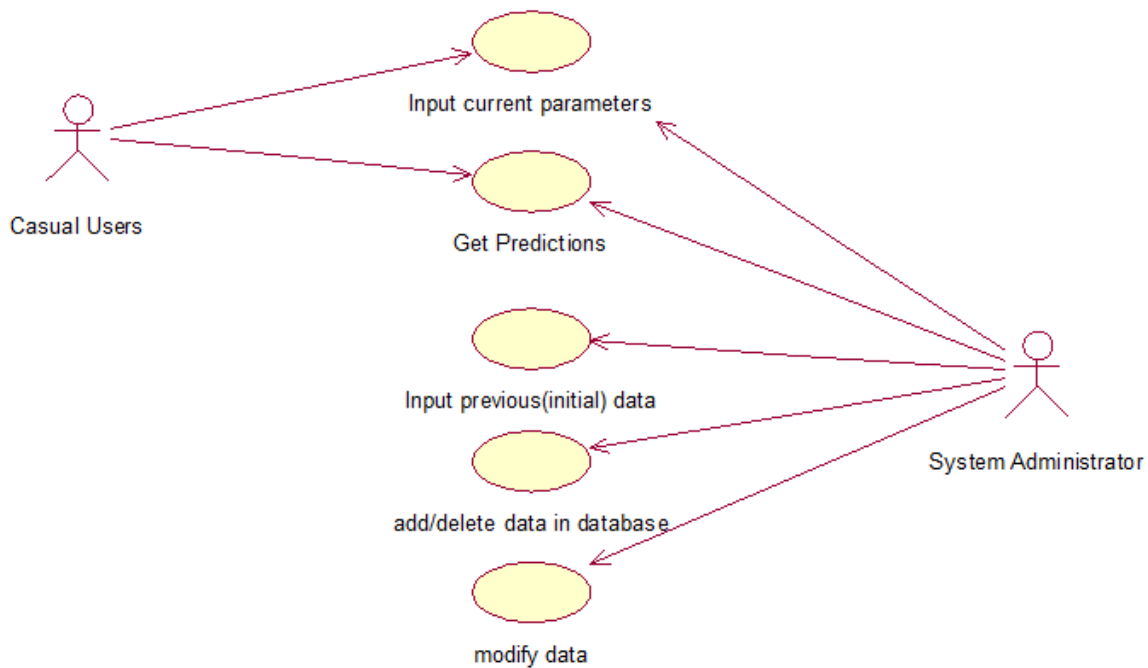
Stakeholders:

- i. Casual Users
- ii. System Administrator

Non-Functional Requirements:

- i. Secure access of confidential data (user's details). SSL or Hashing can be used.
- ii. 24 X 7 availability
- iii. Better component design to get better performance at peak time
- iv. Flexible service-based architecture will be highly desirable for future extension

Use case Diagram:



Functional Requirements:

- i. Should be pluggable for any type of text data available.
- ii. Should have the facility to specify the format of the input structures of the data and adapt accordingly.
- iii. Provide functionalities to access distinct elements or frequent occurrence/s in a specified range which can be dates or values for example.
- iv. Should be able to understand new queries (in predefined format).
- v. Must be scalable.
- vi. Should be able to access data spread across machines.
- vii. Should have the ability to tag unique data sets.
- viii. Response time should be reasonable and provide detail analysis of different task executions done.

Software process model:

The software process model chosen for this project is waterfall model. As we have mentioned our requirements and modules (as in use case diagram) in the abstract clearly and emphasized that these requirements are fixed, and are less likely to change. This is a small-scale university project not as that of a company's or organization's large-scale projects. As all of the process is new to us and we don't have any prior experience in making software, so in order to show that we have learned something from this project, it needs to be well documented and presentable. Documentation is required in each step of waterfall model so this is our advantage.

In Evolutionary development model, process is not visible and system is poorly developed, project is not well documented for each step. University projects should not be left with undocumented steps. We are not using component-based software model as we are not extending any existing project and system does not meet real needs of customers. In Iterative process model it is difficult to prioritize and map the requirement to increments. And it is difficult to identify the common that is needed by all the increments.